PＲTAL
USPTO

Search: ⦿ The ACM Digital Library   ◯ The Guide

firmware archive

SEARCH

THE ACM DIGITAL LIBRARY

⬤ Feedback Report a problem Satisfaction survey

Terms used **firmware** **archive**

Found **6,717** of **192,876**

Sort results by | relevance | ▪
Display results | expanded form | ▪

📀 Save results to a Binder
❓ Search Tips
☐ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200
Best 200 shown

Result page: **1** 2 3 4 5 6 7 8 9 10 next

Relevance scale ☐ ◰ ◩ ◪ ■

1   **Verification of microprogrammed computer architectures in the S*-system: a case study**   ■
W. Damm, G. Dohmen
December 1985 **ACM SIGMICRO Newsletter , Proceedings of the 18th annual workshop on Microprogramming MICRO 18**, Volume 16 Issue 4
Publisher: ACM Press

Full text available: 📄 pdf(1.43 MB)   Additional Information: full citation, abstract, references, citings, index terms

We apply the verification methodology underlying the S*-System[12], [13] to the verification of a hierarchically structured design [16] of an emulation of the instruction-set of a commercially available computer on a commercially available micro-architecture. Based on this case-study, we discuss some aspects of the relation between verification and generation of microcode.

2   **Reliability and security of RAID storage systems and D2D archives using SATA disk drives**   ■
Gordon F. Hughes, Joseph F. Murray
February 2005 **ACM Transactions on Storage (TOS)**, Volume 1 Issue 1
Publisher: ACM Press

Full text available: 📄 pdf(94.82 KB)   Additional Information: full citation, abstract, references, index terms

Information storage reliability and security is addressed by using personal computer disk drives in enterprise-class nearline and archival storage systems. The low cost of these serial ATA (SATA) PC drives is a tradeoff against drive reliability design and demonstration test levels, which are higher in the more expensive SCSI and Fibre Channel drives. This article discusses the tradeoff between SATA which has the advantage that fewer higher capacity drives are needed for a given system storage c ...

Keywords: Disk drive, SATA, SMART, archival storage, failure prediction, secure erase, storage resource management, storage systems architecture

3   **Kernel Korner: Booting the Kernel**   ■
Alessandro Rubini
June 1997 **Linux Journal**
Publisher: Specialized Systems Consultants, Inc.

Full text available: Additional Information:

html(22.08 KB)

full citation, index terms

4 Evaluation of the FACOM ALPHA Lisp machine

M. Yuhara, A. Hattori, M. Niwa, M. Kishimoto, H. Hayashi
June 1986 **ACM SIGARCH Computer Architecture News , Proceedings of the 13th annual international symposium on Computer architecture ISCA '86**, Volume 14 Issue 2

Publisher: IEEE Computer Society Press, ACM Press

Full text available: pdf(508.03 KB)   Additional Information: full citation, abstract, references, index terms

The FACOM ALPHA is the first and only commercially dedicated processor for Lisp and Prolog manufactured in Japan. This paper discusses the evaluation of the FACOM ALPHA for Lisp execution when compared with a general-purpose computer. The CPU use rate of machine instructions and the utilization ratio of the hardware resources are discussed. This paper also explains the interaction between the garbage collector and the virtual memory system, which improves system response time.

5 Extraction of massive instruction level parallelism

Augustus K. Uht
June 1993 **ACM SIGARCH Computer Architecture News**, Volume 21 Issue 3

Publisher: ACM Press

Full text available: pdf(816.46 KB)   Additional Information: full citation, abstract, citings, index terms

Our goal is to dramatically increase the performance of uniprocessors through the exploitation of instruction level parallelism, i.e. that parallelism which exists amongst the machine instructions of a program. Speculative execution may help a lot, but, it is argued, both branch prediction and eager execution are insufficient to achieve performances in speedup factors in the tens (with respect to sequential execution), with reasonable hardware costs.A new form of code execution, *Disjoint Eage ...*

6 Using registers to optimize cross-domain call performance

Paul A. Karger
April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the third international conference on Architectural support for programming languages and operating systems ASPLOS-III**, Volume 17 Issue 2

Publisher: ACM Press

Full text available: pdf(1.30 MB)   Additional Information: full citation, abstract, references, citings, index terms, review

This paper describes a new technique to improve the performance of cross-domain calls and returns in a capability-based computer system. Using register optimization information obtained from the compiler, a trusted linker can minimize the number of registers that must be saved, restored, or cleared when changing from one protection domain to another. The size of the performance gain depends on the level of trust between the calling and called protection domains. The paper presents alternate ...

7 Firmware factory & forth

Brad Eckert
December 1999 **ACM SIGPLAN Notices**, Volume 34 Issue 12

Publisher: ACM Press

Full text available: pdf(373.72 KB)   Additional Information: full citation, citings, index terms

8

Mobile applications: Feedback linking: optimizing object code layout for updates

Carl von Platen, Johan Eker
June 2006 **Proceedings of the 2006 ACM SIGPLAN/SIGBED conference on Language, compilers and tool support for embedded systems LCTES '06**
Publisher: ACM Press
Full text available: pdf(421.85 KB)   Additional Information: full citation, abstract, references, index terms

Firmware over the air (FOTA) is becoming a standard procedure for maintaining and updating wireless embedded systems. To cope with bandwidth and storage constraints this is facilitated using incremental updates based on delta technology, i.e. only the modifications are transmitted. The performance of a FOTA update is highly dependent on the size of the delta, and the type of modifications. Application of a delta update involves mutating the present version, byte by byte, into the new version. Th ...

Keywords: flash memory, incremental software update

9 Design and specification of microprogrammed computer architectures
W. Damm
December 1985 **ACM SIGMICRO Newsletter , Proceedings of the 18th annual workshop on Microprogramming MICRO 18**, Volume 16 Issue 4
Publisher: ACM Press
Full text available: pdf(859.76 KB)   Additional Information: full citation, abstract, references, citings, index terms

This paper presents a hierarchical firmware design method. It allows to structure the design of a microprogrammed (level of a) computer architecture into independently verifiable modules. To specify the behaviour of the system we use the axiomatic architecture description language AADL. We illustrate the design and specification style using an emulation example.

10 A brief survey of current work on network attached peripherals (extended abstract)
Rodney Van Meter
January 1996 **ACM SIGOPS Operating Systems Review**, Volume 30 Issue 1
Publisher: ACM Press
Full text available: pdf(725.39 KB)   Additional Information: full citation, abstract, citings, index terms

Work on network-attached peripherals (NAPs) can be divided into essentially three areas --- device interfaces and protocols, multimedia use and mass storage use. This paper is an extended abstract reviewing some of the current work and provides references and WWW pointers to many of the projects. The impact of this technological advance on operating systems is discussed. The primary purpose of this paper is to broaden understanding of the advantages and pitfalls of NAPs and encourage further res ...

11 Linux and IBM PowerPCs
Daniel Lazenby
March 2000 **Linux Journal**
Publisher: Specialized Systems Consultants, Inc.
Full text available: html(9.58 KB)   Additional Information: full citation, abstract, references, index terms

If you have an RS6000, you could be running Linux.

12 802.1x on Linux with xsupplicant
Matthew Gast
September 2005 **Linux Journal**, Volume 2005 Issue 137
Publisher: Specialized Systems Consultants, Inc.
Full text available: html(17.66 KB)   Additional Information: full citation, abstract, index terms

If you have WPA set up correctly, it's secure. Mick Bauer already did the server, now here's the client side.

13 An architecture with comprehensive facilities of inter-process synchronization and communication

P. Guillier, D. Slosberg

May 1980 **Proceedings of the 7th annual symposium on Computer Architecture**

Publisher: ACM Press

Full text available: pdf(462.26 KB)   Additional Information: full citation, abstract, references, index terms

> In the architecture of the "Level 64" manufactured by CII-Honeywell-Bull and Honeywell Information Systems, processes executing in a central processor are known to the hardware-firmware. They use the same semaphore mechanism as processes executing in an input-output controller. This implies specific data structures recognized by the hardware-firmware and a hardware-firmware dispatching of the central processor resource. Experience in this domain has led to the development of som ...

14 A distributed file system

C. V. Ravi

January 1975 **Proceedings of the 1975 annual conference**

Publisher: ACM Press

Full text available: pdf(450.91 KB)   Additional Information: full citation, abstract, references, index terms

> The design of a file system for a multiprocessor interactive computer system is described. The architecture of the computer system is such that direct access to a resource allocated to a process is not allowed to any other process. Thus, sharing of files can only be done by transmission of copies. The file system is functionally distributed so that different functions run on different processors and some functions are implemented in firmware. Also, there are no global file directories. The ...

15 Firmware structure and architectural support for monitors, vertical migration and user microprogramming

Mamoru Maekawa, Ken Sakamura, Chiaki Ishikawa

March 1982 **ACM SIGARCH Computer Architecture News , ACM SIGPLAN Notices , Proceedings of the first international symposium on Architectural support for programming languages and operating systems ASPLOS-I,** Volume 10 , 17 Issue 2 , 4

Publisher: ACM Press

Full text available: pdf(754.07 KB)   Additional Information: full citation, abstract, references, index terms

> This paper describes firmware and hardware support necessary for constructing easy-to-understand and high performance operating systems including language translators and interpreters. Basic principles are one-to-one correspondence between logical hierarchy and physical hierarchy, and vertical migration. Implementation of monitors in firmware and architectural support for it are discussed, and a sample system is shown. Architectural support for user microprogramming is then discussed and an ...

16 Balanced Multithreading: Increasing Throughput via a Low Cost Multithreading Hierarchy

Eric Tune, Rakesh Kumar, Dean M. Tullsen, Brad Calder

December 2004 **Proceedings of the 37th annual IEEE/ACM International Symposium on Microarchitecture MICRO 37**

Publisher: IEEE Computer Society

Full text available: pdf(197.41 KB)   Additional Information: full citation, abstract

> A simultaneous multithreading (SMT) processor can issue instructions from several threads

every cycle, allowing it to effectively hide various instruction latencies; this effect increases with the number of simultaneous contexts supported. However, each added context on an SMT processor incurs a cost in complexity, which may lead to an increase in pipeline length or a decrease in the maximum clock rate. This paper presents new designs for multithreaded processors which combine a conservative SMT ...

17 Macintosh OS X: a smooth migration

Scott E. Hanselman, Mahmoud Pegah

September 2003 **Proceedings of the 31st annual ACM SIGUCCS conference on User services**

Publisher: ACM Press

Full text available: pdf(208.81 KB)   Additional Information: full citation, abstract, references, index terms

The Ringling School of Art and Design is a fully accredited four year college of visual art and design with a student population of approximately 1000. The Ringling School has achieved national recognition for its large-scale integration of technology into collegiate visual art and design education and maintains a student to computer ratio of better than two to one. Due to the demand for computing power and the requirement for ease of use, we moved our instructional computer laboratories to the ...

Keywords: Macintosh OS X, NFS, NIS, SSH, fonts, migration, network

18 On the design of Always Compatible Instruction Set Architecture(ACISA)

M. R. Bhujade

December 1983 **ACM SIGARCH Computer Architecture News**, Volume 11 Issue 5

Publisher: ACM Press

Full text available: pdf(277.93 KB)   Additional Information: full citation, abstract, references

This note suggests a concept of a new kind of interrupt called GAP to be used in filling gaps in the Architecture of a computing system.

19 The whiteboard: metaphor: a double-edged sword

William Hudson

June 2000 **interactions**, Volume 7 Issue 3

Publisher: ACM Press

Full text available: pdf(506.74 KB)
html(18.98 KB)   Additional Information: full citation, references, citings, index terms

20 The implementation of guaranteed, reliable, secure broadcast networks

Lawrence C. N. Tseung, Keh-Chiang Yu

January 1990 **Proceedings of the 1990 ACM annual conference on Cooperation**

Publisher: ACM Press

Full text available: pdf(802.17 KB)   Additional Information: full citation, abstract, references, citings, index terms

This paper depicts a conceptually simple and easy to implement protocol that provides reliable and secure broadcast/multicast communication. The methodology used in this protocol is surprisingly simple. Three logical nodes are enforced in the network - a Central Retransmitter, a Designated Acknowledger, and a (many when needed) Playback Recorder (s). Through the coordinated service of the three nodes, every user node can be guaranteed to receive all broadcast messages in the correct temporal ...

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next

P⊛RTAL
USPTO

Search: ⦿ The ACM Digital Library   ○ The Guide

| distributed software archive | **SEARCH** |

THE ACM DIGITAL LIBRARY

📧 Feedback Report a problem Satisfaction survey

Terms used **distributed software archive**   Found **72,731** of **192,876**

Sort results by | relevance   ■ | 📗 Save results to a Binder   Try an Advanced Search
Display results | expanded form ■ | ❓ Search Tips   Try this search in The ACM Guide
                                           ☐ Open results in a new window

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown   Relevance scale ☐ ⬜ ◼ ◼ ■

1 Distributed data and immersive collaboration   ■
Daniel A. Reed, Roscoe C. Giles, Charles E. Catlett
November 1997 **Communications of the ACM**, Volume 40 Issue 11
Publisher: ACM Press

Full text available: 📄 pdf(1.36 MB)   Additional Information: full citation, references, citings, index terms, review

2 Knowledge sharing in software engineering: Group awareness in distributed software ■
development
Carl Gutwin, Reagan Penner, Kevin Schneider
November 2004 **Proceedings of the 2004 ACM conference on Computer supported cooperative work**
Publisher: ACM Press
Full text available: 📄 pdf(365.27 KB)   Additional Information: full citation, abstract, references, index terms

Open-source software development projects are almost always collaborative and distributed. Despite the difficulties imposed by distance, these projects have managed to produce large, complex, and successful systems. However, there is still little known about how open-source teams manage their collaboration. In this paper we look at one aspect of this issue: how distributed developers maintain group awareness. We interviewed developers, read project communication, and looked at project artifac ...

Keywords: OSS, collaborative software development, group awareness

3 Networking: Towards a peer2peer world-wide-web for the broadband-enabled user ■
community
Constantine Mantratzis, Mehmet Orgun
October 2004 **Proceedings of the 2004 ACM workshop on Next-generation residential broadband challenges**
Publisher: ACM Press
Full text available: 📄 pdf(243.96 KB)   Additional Information: full citation, abstract, references, index terms

This paper aims to study the concept of a distributed World Wide Web archive that complements the existing WWW and "lives" across a vast Peer-to-Peer network of broadband-connected user nodes.

It proposes the sharing of a web browser's cached data with other peers in an effort to provide an alternative resource to "discontinued" web documents with [normally] short life spans such as video and audio content as well as frequently restructured text pages.

We have based this study o ...

Keywords: distributed world wide web, peer 2 peer

4   Brave new topics 3: advanced methods for medical image retrieval & applications:
    Data grid for large-scale medical image archive and analysis
    H. K. Huang, Aifeng Zhang, Brent Liu, Zheng Zhou, Jorge Documet, Nelson King, L. W. C. Chan
    November 2005 **Proceedings of the 13th annual ACM international conference on Multimedia MULTIMEDIA '05**
    Publisher: ACM Press
    Full text available: pdf(2.03 MB)   Additional Information: full citation, abstract, references, index terms

    Storage and retrieval technology for large-scale medical image systems has matured significantly during the past ten years but many implementations still lack cost-effective backup and recovery solutions. As an example, a PACS (Picture Archiving and Communication system) in a general medical center requires about 40 Terabytes of storage capacity for seven years. Despite many healthcare centers are relying on PACS for 24/7 clinical operation, current PACS lacks affordable fault-tolerance storage ...

    Keywords: PACS, bone age assessment of children, computational services, data grid, fault-tolerance archive, grid computing, image analysis, image data mining

5   Who is an open source software developer?
    Bert J. Dempsey, Debra Weiss, Paul Jones, Jane Greenberg
    February 2002 **Communications of the ACM**, Volume 45 Issue 2
    Publisher: ACM Press
    Full text available: pdf(584.81 KB)   Additional Information: full citation, abstract, references, citings, index
                        html(32.32 KB)   terms

    Profiling a community of Linux developers.

6   Systems: The networked sensor tapestry (NeST): a privacy enhanced software
    architecture for interactive analysis of data in video-sensor networks
    Douglas A. Fidaleo, Hoang-Anh Nguyen, Mohan Trivedi
    October 2004 **Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks**
    Publisher: ACM Press
    Full text available: pdf(674.07 KB)   Additional Information: full citation, abstract, references, index terms

    This paper details the architecture of a test-bed under development for secure sharing, capture, distributed processing, and archiving of surveillance data called the Networked Sensor Tapestry (NeST). The test-bed consists of core software modules including a centralized server, client interface library, a layered XML messaging scheme. Mobile hardware clients are interfaced to the NeST using a Tiny-OS based microcontroller with sensor data collected over a 1-wire data bus. Maintaining subject ...

    Keywords: privacy, surveillance architecture, video-sensor networks

7 <u>Practical extraction techniques for Java</u>

Frank Tip, Peter F. Sweeney, Chris Laffra, Aldo Eisma, David Streeter
November 2002 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 24 Issue 6
Publisher: ACM Press

Full text available: <u>pdf(1.01 MB)</u>   · Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>, <u>review</u>

Reducing application size is important for software that is distributed via the internet, in order to keep download times manageable, and in the domain of embedded systems, where applications are often stored in (Read-Only or Flash) memory. This paper explores extraction techniques such as the removal of unreachable methods and redundant fields, inlining of method calls, and transformation of the class hierarchy for reducing application size. We implemented a number of extraction techniques in < ...

Keywords: Application extraction, call graph construction, class hierarchy transformation, packaging, whole-program analysis

8 <u>Tutorial: Open-source webcasting and media archiving software for e-learning</u>

Ron Baecker, Kelly Rankin
October 2005 **eLearn**, Volume 2005 Issue 10
Publisher: ACM Press
Full text available: <u>html(27.14 KB)</u>
                    <u>Publisher Site</u>   Additional Information: <u>full citation</u>, <u>index terms</u>

9 <u>A distributed scientific data archive using the Web, XML and SQL/MED</u>

Mark Papiani, Jasmin L. Wason, Alistair N. Dunlop, Denis A. Nicole
September 1999 **ACM SIGMOD Record**, Volume 28 Issue 3
Publisher: ACM Press
Full text available: <u>pdf(794.86 KB)</u>   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>citings</u>, <u>index terms</u>

We have developed a web-based architecture and user interface for fast storage, searching and retrieval of large, distributed, files resulting from scientific simulations. We demonstrate that the new DATALINK type defined in the draft SQL Management of External Data Standard can help to overcome problems associated with limited bandwidth when trying to archive large files using the web. We also show that separating the user interface specification from the user interface processing can prov ...

10 <u>Distribution of mathematical software via electronic mail</u>

Jack J Dongarra, Eric Grosse
May 1987 **Communications of the ACM**, Volume 30 Issue 5
Publisher: ACM Press

Full text available: <u>pdf(571.18 KB)</u>   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

A large collection of public-domain mathematical software is now available via electronic mail. Messages sent to "netlib@anl-mcs" (on the Arpanet/CSNET) or to "research! netlib" (on the UNIX® network) wake up a server that distributes items from the collection. The one-line message "send index" causes a library catalog to be sent by return mail.

11 <u>Customized information extraction as a basis for resource discovery</u>
Darren R. Hardy, Michael F. Schwartz

May 1996 **ACM Transactions on Computer Systems (TOCS)**, Volume 14 Issue 2
Publisher: ACM Press

Full text available: pdf(1.91 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>, <u>review</u>

Indexing file contents is a powerful means of helping users locate documents, software, and other types of data among large repositories. In environments that contain many different types of data, content indexing requires type-specific processing to extract information effectively. We present a model for type-specific, user-customizable information extraction, and a system implementation called Essence. This software structure allows users to associate specialized extracti ...

Keywords: Internet, distributed indexing, resource discovery

12 File archive activity in a supercomputing environment

David W. Jensen, Daniel A. Reed
August 1993 **Proceedings of the 7th international conference on Supercomputing**
Publisher: ACM Press

Full text available: pdf(799.48 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

The National Center for Supercomputing Applications (NCSA) at the University of Illinois typifies current, high-performance computing centers that support a diverse user base of both academic and industrial accounts. The data requirements of this user community, coupled with resource constraints, mandate the use of an archival file system. This paper analyzes the spatial and temporal patterns of file archive transactions on the NCSA file archive. Analysis of archive transaction traces shows ...

13 Teams, cognition and methodology issues: Supporting software development as knowledge-intensive and collaborative activity

Yunwen Ye
May 2006 **Proceedings of the 2006 international workshop on Workshop on interdisciplinary software engineering research WISER '06**
Publisher: ACM Press

Full text available: pdf(180.62 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>

Starting from the belief that software development is a human activity, this paper tries to conceptualize software development as a knowledge-intensive design and distributed cognitive activity. This conceptualization leads to the argument that providing support for software developers to engage in knowledge collaboration with external knowledge repositories and peers is essential for software development environments. Technical and social challenges in providing such support are identified, and ...

Keywords: distributed cognition, knowledge collaboration, software development support

14 Monitoring data archives for grid environments

Jason Lee, Dan Gunter, Martin Stoufer, Brian Tierney
November 2002 **Proceedings of the 2002 ACM/IEEE conference on Supercomputing**
Publisher: IEEE Computer Society Press

Full text available: pdf(107.31 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Developers and users of high-performance distributed systems often observe performance problems such as unexpectedly low throughput or high latency. To determine the source of these performance problems, detailed end-to-end monitoring data from applications,

networks, operating systems, and hardware must be correlated across time and space. Researchers need to be able to view and compare this very detailed monitoring data from a variety of angles. To address this problem, we propose a relational ...

15 Mining Software Repositories (MSR): Improving evolvability through refactoring

Jacek Ratzinger, Michael Fischer, Harald Gall

May 2005 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 2005 international workshop on Mining software repositories MSR '05**, Volume 30 Issue 4

Publisher: ACM Press

Full text available: pdf(237.92 KB)   Additional Information: full citation, abstract, references, index terms

Refactoring is one means of improving the structure of existing software. Locations for the application of refactoring are often based on subjective perceptions such as "bad smells", which are vague suspicions of design shortcomings. We exploit historical data extracted from repositories such as CVS and focus on change couplings: if some software parts change at the same time very often over several releases, this data can be used to point to candidates for refactoring. We adopt the concept of b ...

Keywords: change smells, refactoring, software evolution

16 A taxonomy of Data Grids for distributed data sharing, management, and processing

Srikumar Venugopal, Rajkumar Buyya, Kotagiri Ramamohanarao

June 2006 **ACM Computing Surveys (CSUR)**, Volume 38 Issue 1

Publisher: ACM Press

Full text available: pdf(1.70 MB)   Additional Information: full citation, abstract, references, index terms

Data Grids have been adopted as the next generation platform by many scientific communities that need to share, access, transport, process, and manage large data collections distributed worldwide. They combine high-end computing technologies with high-performance networking and wide-area storage management techniques. In this article, we discuss the key concepts behind Data Grids and compare them with other data sharing and distribution paradigms such as content delivery networks, peer-to-peer n ...

Keywords: Grid computing, data-intensive applications, replica management, virtual organizations

17 Distributed transactions in practice

Prabhu Ram, Lyman Do, Pamela Drew

September 1999 **ACM SIGMOD Record**, Volume 28 Issue 3

Publisher: ACM Press

Full text available: pdf(873.01 KB)   Additional Information: full citation, abstract, citings, index terms

The concept of transactions and its application has found wide and often indiscriminate usage. In large enterprises, the model for distributed database applications has moved away from the client-server model to a multi-tier model with large database application software forming the middle tier. The software philosophy of "buy and not build" in large enterprises has had a major influence by extending functional requirements such as transactions and data consistency throughout th ...

18 Interactive execution of distributed algorithms

Mordechai Ben-Ari

August 2001 **Journal on Educational Resources in Computing (JERIC)**

Publisher: ACM Press

Full text available: pdf(75.66 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

DAJ (Distributed Algorithms in Java) is a framework for writing Java programs to implement distributed algorithms. The programs display the data structures at each node and enable the user to interactively construct scenarios. In a learning situation, active interactive execution is preferable to passively watching an animation. Programs have been implemented for commonly taught algorithms, including the Byzantine generals, mutual exclusion, termination, and snapshots. Adding a program for ...

Keywords: Java, byzantine generals, distributed algorithms, mutual exclusion, visualization of algorithms

19 <u>Dissemination of collection wide information in a distributed information retrieval system</u>
Charles L. Viles, James C. French
July 1995 **Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval**
Publisher: ACM Press
Full text available: pdf(841.65 KB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

20 <u>A cooperative approach to support software deployment using the software dock</u>
Richard S. Hall, Dennis Heimbigner, Alexander L. Wolf
May 1999 **Proceedings of the 21st international conference on Software engineering**
Publisher: IEEE Computer Society Press
Full text available: pdf(1.43 MB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Keywords: Java, configuration management, mobile agents, software deployment

Results 1 - 20 of 200   Result page: **1**   <u>2</u>   <u>3</u>   <u>4</u>   <u>5</u>   <u>6</u>   <u>7</u>   <u>8</u>   <u>9</u>   <u>10</u>   <u>next</u>

Useful downloads: <u>Adobe Acrobat</u>   <u>QuickTime</u>   <u>Windows Media Player</u>   <u>Real Player</u>

PꙦRTAL

USPTO

Search: ⦿ The ACM Digital Library   ◯ The Guide

combined source and object code

SEARCH

◀ Feedback Report a problem Satisfaction survey

THE ACM DIGITAL LIBRARY

Terms used **combined** **source** and **object** **code**   Found **102,845** of **192,876**

| Sort results by | relevance ▦ |
| Display results | expanded form ▦ |

❦ Save results to a Binder

🔲 Search Tips

☐ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown   Relevance scale ☐ ▭ ▬ ◼ ◼

---

1 Combining Global Code and Data Compaction   ◼

Bjorn De Sutter, Bruno De Bus, Koen De Bosschere, Saumya Debray
August 2001 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN workshop on Languages, compilers and tools for embedded systems LCTES '01 , Proceedings of the 2001 ACM SIGPLAN workshop on Optimization of middleware and distributed systems OM '01**, Volume 36 Issue 8
Publisher: ACM Press

Full text available: 📄 pdf(191.59 KB)   Additional Information: full citation, abstract, references, citings, index terms

Computers are increasingly being incorporated in devices with a limited amount of available memory. As a result research is increasingly focusing on the automated reduction of program size. Existing literature focuses on either data or code compaction or on highly language dependent techniques. This paper shows how combined code and data compaction can be achieved using a link-time code compaction system that reasons about the use of both code and data addresses. The analyses proposed rely on ...

---

2 Real-time shading   ◼

Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost
August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**
Publisher: ACM Press

Full text available: 📄 pdf(7.39 MB)   Additional Information: full citation, abstract

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...

---

3 Type feedback vs. concrete type inference: a comparison of optimization techniques   ◼
for object-oriented languages

Ole Agesen, Urs Hölzle
October 1995 **ACM SIGPLAN Notices , Proceedings of the tenth annual conference on Object-oriented programming systems, languages, and applications OOPSLA '95**, Volume 30 Issue 10
Publisher: ACM Press

Full text available: 📄 pdf(2.27 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Two promising optimization techniques for object-oriented languages are type feedback (profile-based receiver class prediction) and concrete type inference (static analysis). We directly compare the two techniques, evaluating their effectiveness on a suite of 23 SELF programs while keeping other factors constant.Our results show that both systems inline over 95% of all sends and deliver similar overall performance with one exception: SELF's automatic coercion of machine integer ...

4   An object-oriented model of software configuration management

Hal Render, Roy Campbell
May 1991 **Proceedings of the 3rd international workshop on Software configuration management**
Publisher: ACM Press

Full text available: 📄 pdf(1.21 MB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

5   Combining static and dynamic data in code visualization

David Eng
November 2002 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 2002 ACM SIGPLAN-SIGSOFT workshop on Program analysis for software tools and engineering PASTE '02**, Volume 28 Issue 1
Publisher: ACM Press

Full text available: 📄 pdf(304.66 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

The task of developing, tuning, and debugging compiler optimizations is a difficult one which can be facilitated by software visualization. There are many characteristics of the code which must be considered when studying the kinds of optimizations which can be performed. Both static data collected at compile-time and dynamic runtime data can reveal opportunities for optimization and affect code transformations. In order to expose the behavior of such complex systems, visualizations should inclu ...

Keywords: combining static and dynamic data, intermediate languages, profiling, software understanding, visualization

6   Combination of inheritance hierarchies

Harold Ossher, William Harrison
October 1992 **ACM SIGPLAN Notices , conference proceedings on Object-oriented programming systems, languages, and applications OOPSLA '92**, Volume 27 Issue 10
Publisher: ACM Press

Full text available: 📄 pdf(1.85 MB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

7   Reverse engineering framework reuse interfaces

Jukka Viljamaa
September 2003 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 9th European software engineering conference held jointly with 11th ACM SIGSOFT international symposium on Foundations of software engineering ESEC/FSE-11**, Volume 28 Issue 5
Publisher: ACM Press

Full text available: 📄 pdf(313.32 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>

Object-oriented application frameworks provide an established way of reusing the design and implementation of applications in a specific domain. Using a framework for creating applications is not a trivial task, however, and special tools are needed for supporting the process. Tool support, in turn, requires explicit specification of the *reuse interfaces* of frameworks. Unfortunately these specifications typically become quite extensive and complex for non-trivial frameworks. In thi ...

Keywords: documentation, formal concept analysis, framework, pattern, reuse, reverse engineering

8   The type system for object initialization in the Java bytecode language

Stephen N. Freund, John C. Mitchell

November 1999 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 21 Issue 6

Publisher: ACM Press

Full text available: 📄 pdf(394.88 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

In the standard Java implementation, a Java language program is compiled to Java bytecode. This bytecode may be sent across the network to another site, where it is then executed by the Java Virtual Machine. Since bytecode may be written by hand, or corrupted during network transmission, the Java Virtual Machine contains a bytecode verifier that performs a number of consistency checks before code is run. These checks include type correctness and, as illus-trated by previous attacks on the J ...

Keywords: Java, bytecode languages, object initialization, type checking

9   Jeroo: a tool for introducing object-oriented programming

Dean Sanders, Brian Dorn

January 2003 **ACM SIGCSE Bulletin , Proceedings of the 34th SIGCSE technical symposium on Computer science education SIGCSE '03**, Volume 35 Issue 1

Publisher: ACM Press

Full text available: 📄 pdf(213.19 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Jeroo is a tool that has been developed to help students in beginning programming courses learn the semantics of fundamental control structures, learn the basic notions of using objects to solve problems, and learn to write methods that support a functional decomposition of the task. Jeroo is similar to Karel the Robot and its descendants, but has a narrower scope than Karel's descendants and has a syntax that provides a smoother transition to either Java or C++. Jeroo has been class tested at N ...

Keywords: CS1, microworlds, objects-first, pedagogy

10   A type system for object initialization in the Java bytecode language

Stephen N. Freund, John C. Mitchell

October 1998 **ACM SIGPLAN Notices , Proceedings of the 13th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications OOPSLA '98**, Volume 33 Issue 10

Publisher: ACM Press

Full text available: 📄 pdf(1.91 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

In the standard Java implementation, a Java language program is compiled to Java bytecode. This bytecode may be sent across the network to another site, where it is then

interpreted by the Java Virtual Machine. Since bytecode may be written by hand, or corrupted during network transmission, the Java Virtual Machine contains a *bytecode verifier* that performs a number of consistency checks before code is interpreted. As illustrated by previous attacks on the Java Virtual Machine, these test ...

11 Automatic-programming-language translation through syntactical analysis

Robert S. Ledley, James B. Wilson
March 1962 **Communications of the ACM**, Volume 5 Issue 3

Publisher: ACM Press

Full text available: pdf(1.22 MB)   Additional Information: full citation, references, citings

12 MemSpy: analyzing memory system bottlenecks in programs

Margaret Martonosi, Anoop Gupta, Thomas Anderson
June 1992 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1992 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems SIGMETRICS '92/PERFORMANCE '92,**
Volume 20 Issue 1

Publisher: ACM Press

Full text available: pdf(1.57 MB)   Additional Information: full citation, abstract, references, citings, index terms

To cope with the increasing difference between processor and main memory speeds, modern computer systems use deep memory hierarchies. In the presence of such hierarchies, the performance attained by an application is largely determined by its memory reference behavior—if most references hit in the cache, the performance is significantly higher than if most references have to go to main memory. Frequently, it is possible for the programmer to restructure the data or code to achieve bet ...

13 Large meshes and GPU programming: Shader algebra

Michael McCool, Stefanus Du Toit, Tiberiu Popa, Bryan Chan, Kevin Moule
August 2004 **ACM Transactions on Graphics (TOG)**, Volume 23 Issue 3

Publisher: ACM Press

Full text available: pdf(355.37 KB)   Additional Information: full citation, abstract, references, citings, index
.mov(23:56 MIN)   terms

An algebra consists of a set of objects and a set of operators that act on those objects. We treat shader programs as first-class objects and define two operators: connection and combination. Connection is functional composition: the outputs of one shader are fed into the inputs of another. Combination concatenates the input channels, output channels, and computations of two shaders. Similar operators can be used to manipulate streams and apply computational kernels expressed as shaders to strea ...

Keywords: graphics hardware, real-time rendering, shader programming

14 An evolutionary approach to constructing effective software reuse repositories

Scott Henninger
April 1997 **ACM Transactions on Software Engineering and Methodology (TOSEM)**,
Volume 6 Issue 2

Publisher: ACM Press

Full text available: pdf(662.79 KB)   Additional Information: full citation, abstract, references, citings, index
                                                                            terms, review

Repositories for software reuse are faced with two interrelated problems: (1) acquiring the knowledge to initially construct the repository and (2) modifying the repository to meet

· the evolving and dynamic needs of software development organizations. Current software repository methods rely heavily on classification, which exacerbates acquistition and evolution problems by requiring costly classification and domain analysis efforts before a repository can be used effectively, This article o ...

Keywords: component repositories, information retrieval, software reuse

15 Session 5: test execution: From daikon to agitator: lessons and challenges in building ■ a commercial tool for developer testing
Marat Boshernitsan, Roongko Doong, Alberto Savoia
July 2006 **Proceedings of the 2006 international symposium on Software testing and analysis ISSTA '06**
Publisher: ACM Press
Full text available: ⬚ pdf(1.09 MB)   Additional Information: full citation, abstract, references, index terms

Developer testing is of one of the most effective strategies for improving the quality of software, reducing its cost, and accelerating its development. Despite its widely recognized benefits, developer testing is practiced by only a minority of developers. The slow adoption of developer testing is primarily due to the lack of tools that automate some of the more tedious and time-consuming aspects of this practice. Motivated by the need for a solution, and helped and inspired by the research in ...

Keywords: automated testing tools, developer testing, dynamic invariant detection, software agitation, technology transfer, test-input generation, unit testing

16 *AdJava*: automatic distribution of Java applications   ■
Mohammad M. Fuad, Michael J. Oudshoorn
January 2002 **Australian Computer Science Communications , Proceedings of the twenty-fifth Australasian conference on Computer science - Volume 4 ACSC '02**, Volume 24 Issue 1
Publisher: Australian Computer Society, Inc., IEEE Computer Society Press
Full text available: ⬚ pdf(1.27 MB)   Additional Information: full citation, abstract, references, citings, index terms

The majority of the world's computing resources remains idle most of the time. By using this resource pool, an individual computation may be completed in a fraction of time required to run the same computation on a single machine. However, distributing a program over a number of machines proves to be a tedious and difficult job. This paper introduces a system, called AdJava, which harnesses the computing power of these under-utilized heterogeneous computers by automatically distributing the user ...

Keywords: distributed programming, software agents.

17 Courses: An introduction to sketch-based interfaces   ■
Joseph LaViola, Randall Davis, Takeo Igarashi
July 2006 **Material presented at the ACM SIGGRAPH 2006 conference SIGGRAPH '06**
Publisher: ACM Press
Full text available: ⬚ pdf(31.58 MB)   Additional Information: full citation, abstract

Sketch-based interfaces are a natural, pencil-and-paper-like approach to interacting with a variety of applications, including conceptual modeling, animation, and note-taking systems. This course offers an in-depth discussion of sketch-based interface design, ranging from simple gestural commands to complex sketch-understanding systems. Attendees will learn how these interfaces are designed and how to develop their own.

18 Object combining: A new aggressive optimization for object intensive programs

Ronald Veldema, J. H. Ceriel, F. H. Rutger, E. Henri

November 2002 **Proceedings of the 2002 joint ACM-ISCOPE conference on Java Grande**

Publisher: ACM Press

Full text available: pdf(99.27 KB)   Additional Information: full citation, abstract, references, index terms

Object combining tries to put objects together that have roughly the same life times in order to reduce strain on the memory manager and to reduce the number of pointer indirections during a program's execution. Object combining works by appending the fields of one object to another, allowing allocation and freeing of multiple objects with a single heap (de)allocation. Unlike object *inlining*, which will only optimize objects where one has a (unique) pointer to another, our optimization al ...

Keywords: Java, garbage collection, object management

19 Finding application errors and security flaws using PQL: a program query language

Michael Martin, Benjamin Livshits, Monica S. Lam

October 2005 **ACM SIGPLAN Notices , Proceedings of the 20th annual ACM SIGPLAN conference on Object oriented programming, systems, languages, and applications OOPSLA '05**, Volume 40 Issue 10

Publisher: ACM Press

Full text available: pdf(317.30 KB)   Additional Information: full citation, abstract, references, citings, index terms

A number of effective error detection tools have been built in recent years to check if a program conforms to certain design rules. An important class of design rules deals with sequences of events asso-ciated with a set of related objects. This paper presents a language called PQL (Program Query Language) that allows programmers to express such questions easily in an application-specific context. A query looks like a code excerpt corresponding to the shortest amount of code that would violate a ...

Keywords: SQL injection, bug finding, pattern matching, program traces, resource leaks, web applications

20 Object code optimization

Edward S. Lowry, C. W. Medlock

January 1969 **Communications of the ACM**, Volume 12 Issue 1

Publisher: ACM Press

Full text available: pdf(1.30 MB)   Additional Information: full citation, abstract, references, citings, index terms

Methods of analyzing the control flow and data flow of programs during compilation are applied to transforming the program to improve object time efficiency. Dominance relationships, indicating which statements are necessarily executed before others, are used to do global common expression elimination and loop identification. Implementation of these and other optimizations in OS/360 FORTRAN H are described.

Keywords: FORTRAN, System/360, compilers, data flow analysis, dominance, efficiency, graph theory, loop structure, machine instructions, object code, optimization, redundancy elimination, register assignment

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next